IB/2004/04349

INVESTOR IN PEOPLE

The Patent Office
Concept House
Cardiff Road
Newport
South Wales
NP10 8QQ

I, the undersigned, being an officer duly authorised in accordance with Section 74(1) and (4) of the Deregulation & Contracting Out Act 1994, to sign and issue certificates on behalf of the Comptroller-General, hereby certify that annexed hereto is a true copy of the documents as originally filed in connection with the patent application identified therein.

In accordance with the Patents (Companies Re-registration) Rules 1982, if a company named in this certificate and any accompanying documents has re-registered under the Companies Act 1980 with the same name as that with which it was registered immediately before re-registration save for the substitution as, or inclusion as, the last part of the name of the words "public limited company" or their equivalents in Welsh, references to the name of the company in this certificate and any accompanying documents shall be treated as references to the name with which it is so re-registered.

In accordance with the rules, the words "public limited company" may be replaced by p.l.c., plc, P.L.C. or PLC.

Re-registration under the Companies Act does not constitute a new legal entity but merely subjects the company to certain additional company law rules.

Signed Stephen Hordley

Dated 13 January 2005

PRIORITY DOCUMENT
SUBMITTED OR TRANSMITTED BUT NOT IN COMPLIANCE WITH RULE 17.1(a) OR (b)

ents Form 1/77

ct 1977

The Patent Office

THE PATENT OFFICE
G
11 DEC 2003
LONDON

EC03 E03879[illegible]
/7700.00-0328794.3 ACCOUNT CHA

Request for grant of a patent

(See the notes on the back of this form. You can also get an explanatory leaflet from the Patent Office to help fill in this form)

The Patent Office

Cardiff Road
Newport
Gwent NP9 1RH

Your reference **SC13079EI/CJH/GBRI/JT/KUTZ**

Patent application number
(The Patent Office will fill in this part) 11 DEC 2003 0328794.3

Full name, address and postcode of the or of each applicant *(underline all surnames)*

MOTOROLA, INC
1303 EAST ALGONQUIN ROAD,
SCHAUMBURG, ILLINOIS 60196,
U.S.A.

00615336004
Patents ADP number *(if you know it)* **08281107001**

If the applicant is a corporate body, give the country/state of its incorporation **U.S.A.** **DELAWARE**

Title of the invention **A DECODER**

Name of your agent *(if you have one)*

"Address for service" in the United Kingdom to which all correspondence should be sent *(including the postcode)*

HARRISON, CHRISTOPHER
EUROPEAN INTELLECTUAL PROPERTY DEPARTMENT
MIDPOINT
ALENCON LINK
BASINGSTOKE
HAMPSHIRE RG21 7PL
UK
ADP NO. 00001180006

Patents ADP number *(if you know it)* 07957749001

6. If you are declaring priority from one or more earlier patent applications, give the country and the date of filing of the or of each of these earlier applications and *(if you know it)* the or each application number

Country	Priority application number *(if you know it)*	Date of filing *(day / month / year)*

7. If this application is divided or otherwise derived from an earlier UK application, give the number and the filing date of the earlier application

Number of earlier application	Date of filing *(day / month / year)*

8. Is a statement of inventorship and of right to grant of a patent required in support of this request? *(Answer 'Yes' if:*

a) any applicant named in part 3 is not an inventor, or
b) there is an inventor who is not named as an applicant, or
c) any named applicant is a corporate body.
See note (d))

☒ YES ☐ NO

Patents Form 1/77

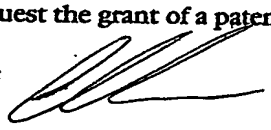
9. Enter the number of sheets for any of the following items you are filing with this form. Do not count copies of the same document

Continuation sheets of this form	
Description	11
Claim(s)	2
Abstract	1
Drawing(s)	3 +3

10. If you are also filing any of the following, state how many against each item.

Priority documents	
Translations of priority documents	
Statement of inventorship and right to grant of a patent *(Patents Form 7/77)*	3
Request for preliminary examination and search *(Patents Form 9/77)*	1
Request for substantive examination *(Patents Form 10/77)*	1
Any other documents *(please specify)*	1 x FEE SHEET

11. I/We request the grant of a patent on the basis of this application.

Signature  Date 12/10/2003

12. Name and daytime telephone number of person to contact in the United Kingdom

HARRISON, CHRISTOPHER
Eliza LAURIE 01252-790763

Warning

After an application for a patent has been filed, the Comptroller of the Patent Office will consider whether publication or communication of the invention should be prohibited or restricted under Section 22 of the Patents Act 1977. You will be informed if it is necessary to prohibit or restrict your invention in this way. Furthermore, if you live in the United Kingdom, Section 23 of the Patents Act 1977 stops you from applying for a patent abroad without first getting written permission from the Patent Office unless an application has been filed at least 6 weeks beforehand in the United Kingdom for a patent for the same invention and either no direction prohibiting publication or communication has been given, or any such direction has been revoked.

Notes

a) *If you need help to fill in this form or you have any questions, please contact the Patent Office on 0645 500505.*

b) *Write your answers in capital letters using black ink or you may type them.*

c) *If there is not enough space for all the relevant details on any part of this form, please continue on a separate sheet of paper and write "see continuation sheet" in the relevant part(s). Any continuation sheet should be attached to this form.*

d) *If you have answered 'Yes' Patents Form 7/77 will need to be filed.*

e) *Once you have filled in the form you must remember to sign and date it.*

f) *For details of the fee and ways to pay please contact the Patent Office.*

A DECODER

The present invention relates to a decoder.

Wireless communication systems are widely deployed to provide various types of communications such as voice and data. One such system is wideband code division multiple access WCDMA, which has been adopted in various competing wireless communication standards, for example 3rd generation partnership project 3GPP and 3GPP2.

To overcome data corruption that can occur during RF transmission the different wireless communication standards typically include some form of channel coding. For example, WCDMA standards typically require that a WCDMA receiver decode a mixture of turbo encoded and viterbi encoded data streams, where viterbi encoded channels are typically used for time critical data.

In particular the 3GPP standard has specified a high speed data packed access HSDPA sub-system that has two physical channels in the downlink direction; a data channel and a control channel in which turbo coding is used to encode the data channel and convolutional coding is used to encode the control channel.

A turbo encoder uses a first convolutional encoder to encode information bits (i.e. systematic bits) within a packet to generate a first sequence of parity bits (i.e. parity 1 bits) in parallel to the interleaver shuffling the information bits, where the shuffled information bits are encoded by a second encoder to generate a second sequence of parity bits (i.e. parity 2 bits). The information bits and the parity bits in the first and second sequence are then modulated and transmitted to a receiver.

The information bits and the first and second sequence of parity bits are received by a receiver and decoded by a turbo decoder.

For turbo encoded data a turbo decoder initially stores the received information bits and the parity bits in the first and second sequence in a buffer. Initially, the information bits and the first sequence of parity bits from the first convolutional encoder are retrieved from the buffer and decoded by a first decoder (i.e. a first soft in soft out SISO decoder), using a log-MAP algorithm, to provide 'extrinsic' information (i.e. a-posteriori data) indicative of adjustments in the confidence in the detected values for the information bits. Intermediate results (i.e. a-priori) that include the extrinsic information from the first decoder are then stored in the buffer in an interleaved order matching the code interleaving used at the transmitter.

The intermediate results, the information bits and the second sequence of parity bits from the second encoder are retrieved from the buffer and decoded by a second decoder (i.e. a second SISO decoder) to provide extrinsic information indicative of further adjustments in the confidence in the detected values for the information bits. Intermediate results that comprise the extrinsic information from the second decoder (i.e. a second SISO decoder) are then stored in the buffer in a deinterleaved order complementary to the code interleaving performed at the transmitter. The intermediate results are used in a next decoding iteration performed by the turbo decoder. The turbo decoder performs a predetermined number of decoding iterations before producing a decision on the value of the decoded information bit.

A viterbi decoder is used to decode convolutional encoded data using a viterbi algorithm.

The basic transmission unit in a HSDPA sub-system is called a time transmission interval TTI where each TTI spans 2ms and contains three identical time periods called slots.

As shown in figure 1, for each TTI transmitted in the data channel 100 there is a corresponding TTI in the control channel 200 that starts 2 slots before the beginning of the associated data channel TTI.

The control data is divided into two parts. The first part 102, which contains information required for the demodulation of the corresponding data channel TTI, is transmitted in the first slot of the control channel TTI. The second part 103, which contains data required for the channel decoding of the corresponding data channel TTI, is transmitted in the second and third slots of the control channel TTI.

There is a period of one slot to decode the first part of the control channel before the decoded data is required for the demodulation of the data channel. Similarly, there is a two slot period to decode the second part of the control channel before the decoded data is required for decoding of the data channel. This arrangement results in severe timing restrictions on the decoding of the control channel part.

One solution to this problem has been the use of a separate turbo decoder for decoding the turbo encoded channels and a separate viterbi decoder for decoding the convolutional encoded channels; however this results in increased cost and size of a receiver.

It is desirable to improve this situation.

In accordance with a first aspect of the present invention there is provided a decoder according to claim 1.

This provides the advantage of allowing a single decoder to support the decoding of a turbo encoded channel and a convolutional encoded channel.

An embodiment of the invention will now be described, by way of example, with reference to the drawings, in which:

Figure 1 illustrates the structure of a known HS-DPA data channel and control channel;

Figure 2 illustrates a WCDMA receiver according to an embodiment of the present invention;

Figure 3 illustrates a decoder according to an embodiment of the present invention;

Figure 4 illustrates a first memory structure according to an embodiment of the present invention;

Figure 5 illustrates a second memory structure according to an embodiment of the present invention.

Figure 2 shows a WCDMA receiver 200 having a memory module 201 (e.g. a buffer), a controller 202 and a decoder 203.

The memory module 201 has a first input for receiving encoded data, a second input for receiving decoded data from the decoder 203, a first output for outputting decoded data, and a second output for providing stored data to the

decoder 203. Additionally, the memory module 201 is coupled to the controller 202 to allow the controller 202 to control the flow of data into and out of the memory module 201.

The memory module 201 has four storage areas 204, 205, 206, 207; one storage area 204 for storing received viterbi encoded data (e.g. encoded data received from the HS-DPA control channel), a second storage area 205 for storing received turbo encoded data (e.g. encoded data received from the HS-DPA data channel), a third storage area 206 for storing decoded data received from the decoder, and a fourth storage area 207 for storing viterbi decoded data.

The decoder 203, which is described in detail below, includes a decoding module 208 arranged to support the decoding of both turbo encoded data and convolutional encoded data and an internal memory module 209, where the internal memory module 209 is relatively small with a relatively high bandwidth. The decoding module 208 is arranged to function as a soft input soft output SISO decoder when performing turbo decoding and a hard decoder when performing viterbi decoding.

The decoder 203 is coupled to the controller 202 via a control line to allow the controller 202 to control the operation of the decoder.

An example of the decoding of turbo encoded data and viterbi encoded data will know be described.

The controller 202 initiates the loading of received turbo encoded data into the second storage area 205 of the memory module 201 and starts the turbo decoding process by issuing a 'turbo start' command to the decoder 203. The decoder 203 retrieves systematic bits and the parity 1 bits from the memory module 201 and performs a SISO turbo decoding stage, which corresponds to the

first decoder process in a turbo decoder, to generate 'extrinsic' information (i.e. a-posterior data), which is stored in the third storage area 206 of the memory module 201. The decoder 203 then retrieves the extrinsic information, which is read out in an interleaved order to provide a-prior data, the systematic bits and the parity 2 bits from the memory module 201 and performs a SISO turbo decoding stage, which corresponds to the second decoding process, thereby completing one turbo decoding iteration. The results from this SISO turbo decoding iteration are stored in the third storage area 206 of the memory module 201.

When part 1 of the HSDPA control channel has been received the controller 202 issues a 'viterbi request' command to the decoder 203. In response the decoder 203 finishes the SISO turbo decoding stage that is currently being processed and notifies the controller 202, via a 'viterbi acknowledge' command, when the SISO turbo decoding stage has been complete, while storing the 'extrinsic' data in the third storage area 206 in the memory module 201.

The controller 202 initiates the loading of received convolutional encoded data into the first storage area 204 of the memory module 201 and starts the viterbi decoding process by issuing a 'viterbi start' command to the decoder 203. The decoder 203 retrieves the viterbi encoded data from the first storage area 204 and performs viterbi decoding. When the decoder 203 has completed the viterbi decoding the decoder 203 notifies the controller 202, via a 'decode end' command, and the decoded viterbi data is stored in the fourth storage area 207 of the memory module 201.

The controller 202 then instructs the decoder 203 to continue the decoding of the turbo encoded data (i.e. instructs the decoder 203 to continue performing decoding iterations on the data stored in the third storage area 206 of the memory module 201), via a 'turbo continue' command. The decoder 203 retrieves the data stored in the third storage area 206 of the memory module 201 and continues the turbo decoding process from the last SISO turbo decoding stage performed.

When all the turbo decoding iterations have been complete for a given data slot the decoder 203 notifies the controller 202, via a 'decode end' command, and the decoded data is stored in the third storage area 206 for further processing by other modules (not shown) within the receiver.

If further encoded data within HSDPA data channel slots is available the controller 202 initiates the loading of the received turbo encoded data into the second storage area 205 of the memory module 201 and initiates the turbo decoding process, as described above.

When part 2 of the HSDPA control channel is received the controller 202 initiates the decoding of this data, as described above.

The switching of the decoder 203 between turbo decoding of the data channel and viterbi decoding of the control channel is arranged to continue while the HSDPA data and control channels are being received.

Figure 3 shows the decoder 203 and the memory module 209. The decoder 203 includes a memory interface 301, a branch metric arithmetic logic unit ALU 302, a butterfly/survivor path ALU 303 arranged to calculate 4 butterflies in a single cycle, a MAX unit 304 and a temporary memory 305.

The memory interface 301 is used to interface the decoder 203 to the memory module 201 with the branch metric ALU 302, the MAX unit 304 and the temporary memory 305 being coupled to the memory interface 301. The branch metric ALU 302 is also coupled to the butterfly/survivor path ALU 303, which in turn is coupled to the internal memory module 209. The MAX unit 304 is also coupled to the internal memory module 209 and the temporary memory 305.

During the decoding process the branch metric ALU 302 receives encoded data from the memory module 201 via the memory interface 301. For example, during turbo decoding turbo encoded data is received from the second storage area 205 of the memory module 201 and during viterbi decoding convolutional encoded data is received from the first storage area 204 of the memory module 201.

The branch metric ALU 302 performs branch metric calculations on the received encoded data and provides the calculated branch metrics to the butterfly/survivor path ALU 303.

During turbo decoding the butterfly/survivor path ALU 303 calculates forward recursion state metrics (i.e. alphas α) and backward recursion state metrics (i.e. betas β) while during viterbi decoding the butterfly/survivor path ALU 303 calculates path metrics and updated survivor path metrics. The number of code states for turbo codes and viterbi codes will typically be different, for example for the HSDPA channel there are 8 states for turbo codes (i.e. the constraint length for the turbo code is 4) and 256 states for viterbi codes (i.e. the constraint length for the viterbi code is 9). This is supported by the butterfly/survivor path ALU 303 by the butterfly/survivor path ALU 303 calculating 8 states of a turbo code in parallel in 1 cycle and 256 states of a viterbi code in 32 cycles (i.e. 8 states in parallel 32 times). As described below, the internal memory module 209 is used to allow the butterfly/survivor path ALU 303 to divide the path metric calculation over 32 cycles by storing both the old and new path metrics in memory (i.e. the path metrics for the previous and current cycle).

The butterfly/survivor path ALU 303 results are stored in the internal memory module 209. The internal memory module 209 within this embodiment is configured to have two separate modules, where each module has its own address bus, thereby allowing memory accesses to different addresses between the two modules within a single clock cycle. For the purposes of this embodiment

the first module has been designated memory group A and the second module has been designated memory group B. Each memory group is one kilobyte in size having 64 lines (i.e. memory addresses) of 16 bytes each, as described below. The internal memory module 209 is small and can be regarded as a window to a code segment stored in the memory module 201, where the decoding of the code segment is effectively a series of decoding operations on the window data where the window 'slides' over the code segment.

The internal memory module 209 is arranged to store path metrics for a previous cycle, when the decoder 203 is in viterbi mode, which is feedback to the butterfly/survivor path ALU 303 to allow the path metrics for the next cycle to be calculated. When the decoder 203 is in turbo mode, the internal memory module 209 is arranged to store all the state metrics of one window for use by the MAX* unit 304.

Once the decoder 203 has performed the decoding of the states in viterbi decoding the calculated survivor path metrics for the received convolutional encode data is passed from the internal memory 209 through the MAX unit 304 and stored in the fourth storage area 207 of the memory module 201.

When the decoder 203 is performing turbo decoding the calculated forward recursion state metrics and backward recursion state metrics are passed to the MAX unit 304, via the internal memory module 209. Additionally, the temporary memory 305 is used to store extrinsic information that is read out of the third storage area 206 in interleaved order (i.e. a-priori data) which is also provided to the MAX unit 304 to allow the MAX unit 304 to determine new a-posterior data for the current SISO turbo decoding stage. The a-posterior data calculated by the MAX unit 304 is stored in the third storage area 206 of the memory module 201.

As described above, the decoder 203 can be controlled to switch between turbo decoding and viterbi decoding, however, as the contents of the internal

memory module 209 are written over when switching from one decoding mode to another decoding mode the switching from one decoding mode to another decoding mode only occurs once a SISO decoding stage has been complete and the SISO decoding stage results have been stored in the memory module or viterbi decoding has been complete.

Figure 4 illustrates the data storage within the internal memory module 209 when being used for turbo decoding. Memory group A and memory group B together are used to store 128 turbo decoding stages, where a stage corresponds to the 8 state metrics of a turbo code trellis (i.e. the forward recursive metrics or backward recursive metrics calculated by the butterfly/survivor path ALU).

For each cycle two stages of data is read or written from/to the memory.

Figure 5 illustrates the data storage within the internal memory module 209 when being used for viterbi decoding. As the butterfly/survivor path ALU 303 is arranged to calculate four butterflies in a single cycle (i.e. eight states of the 256 states) the internal memory module is divided into two parts, part A of group A, part B of group B and part A of group B, part B of group A, to avoid corrupting stage i-1 data stored in the internal memory module with the results of the stage i data, until all the states of stage i have been calculated.

For example, the path metrics and survivor paths calculated by the butterfly/survivor ALU 303 for the current stage are read from the part A of group A, part B of group B having been written in during calculation of the previous stage, where the inputs for the first cycle butterflies correspond to i= 0, i = 1, i = 2, i = 3, where i takes values from 0 to 127 (i.e. 128 butterflies) and the input states read from memory are states 0 to 3 and 128 to 131. The new path metrics and survivor path calculations generated by the butterfly/survivor path ALU 303 are written to part A of group B, part B of group A and correspond to states 0 to 3 and 4 to 7. During calculations for the next stage the roles of part A of group A, part B

of group B and part A of group B, part B of group A of the internal memory module 209 are exchanged (i.e. part A of group A, part B of group B is used for writing the next stage and part A of group B, part B of group A is used to read the current stage data). This process continues until the 256 states have been calculated. With this ordering each read and write operation will involve 4 states from part A of group A, part B of group B and four from part A of group B, part B of group A. Thus, the ordering of the states within the two memory groups allows data to be read from each of the two memory groups or to be written to each of the two memory groups in a single cycle.

It will be apparent to those skilled in the art that the disclosed subject matter may be modified in numerous ways and may assume many embodiments other than the preferred forms specifically set out as described above, for example the decoder could be configured to have one ALU for performing forward recursion and another ALU for performing backward recursion such that the two operations could be performed in parallel.

CLAIMS

1. A decoder comprising a decoding element arranged to operate in a first mode for decoding a turbo encoded data stream and in a second mode for decoding a viterbi encoded data stream, wherein the decoding element is responsive to a first control signal for switching from the first mode to the second mode during decoding of a turbo code block and responsive to a second control signal for switching from the second mode to the first mode to allow continued decoding of the turbo code block.

2. A decoder according to claim 1, wherein the decoder element is arranged to store data generated during the decoding of the turbo code block.

3. A decoder according to claim 2, wherein the decoder element arranged to retrieve the stored data generated during the decoding of the turbo code block to allow continued decoding of the turbo code block.

4. A decoder according to claim 1, wherein the decoding element is arranged to switch from the first mode to the second mode after an iteration of the decoding of the turbo code block has been complete.

5. A decoder according to claim 1, wherein the decoder element comprises a first logic element that is arranged to calculate forward recursion/backward recursion metrics for a turbo encoded data stream when the decoder element is operating in the first mode and to calculate path metrics and survivor path metrics for a viterbi encoded data stream when the decoder element is operating in the second mode.

6. A decoder according to claim 2, wherein the decoder element further comprises a second logic unit that is arranged to calculate a posteriori data for a turbo encoded data stream using the forward recursion/backward recursion metrics generated by the first logic unit.

7. A decoder according to claim 2 or 3, wherein the decoder element further comprises a memory that is arranged to store forward recursion/backward recursion metrics generated by the first logic unit when the decoder element is operating in the first mode and to store path metrics and survivor path metrics generated by the first logic unit when the decoder element is operating in the second mode.

8. A decoder substantially as herein described with reference to the accompanying figures.

ABSTRACT

A DECODER

A decoder comprising a decoding element arranged to operate in a first mode for decoding a turbo encoded data stream and in a second mode for decoding a viterbi encoded data stream, wherein the decoding element is responsive to a first control signal for switching from the first mode to the second mode during decoding of a turbo code block and responsive to a second control signal for switching from the second mode to the first mode to allow continued decoding of the turbo code block.

(Figure 4)

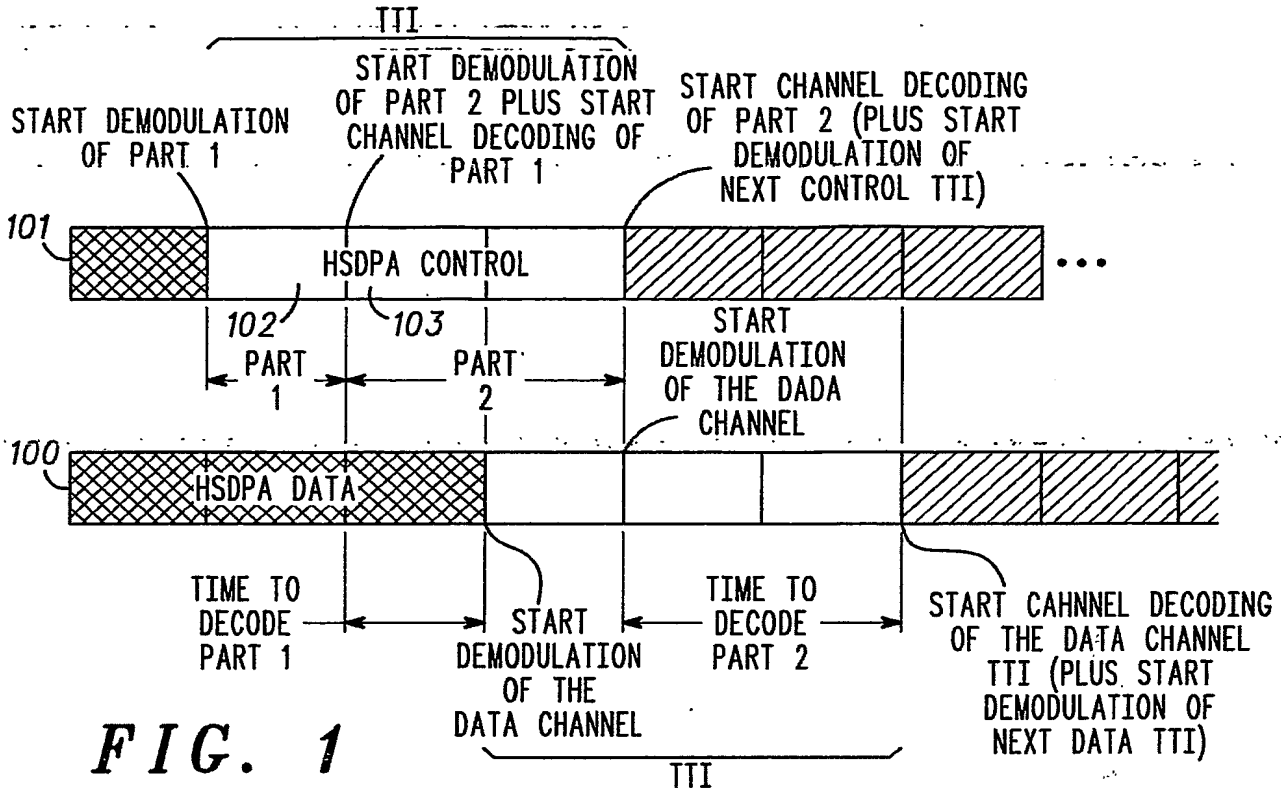

TTI
START DEMODULATION OF PART 1
START DEMODULATION OF PART 2 PLUS START CHANNEL DECODING OF PART 1
START CHANNEL DECODING OF PART 2 (PLUS START DEMODULATION OF NEXT CONTROL TTI)
101
HSDPA CONTROL
102
103
PART 1
PART 2
START DEMODULATION OF THE DADA CHANNEL
100
HSDPA DATA
TIME TO DECODE PART 1
START DEMODULATION OF THE DATA CHANNEL
TIME TO DECODE PART 2
START CAHNNEL DECODING OF THE DATA CHANNEL TTI (PLUS START DEMODULATION OF NEXT DATA TTI)
TTI

FIG. 1

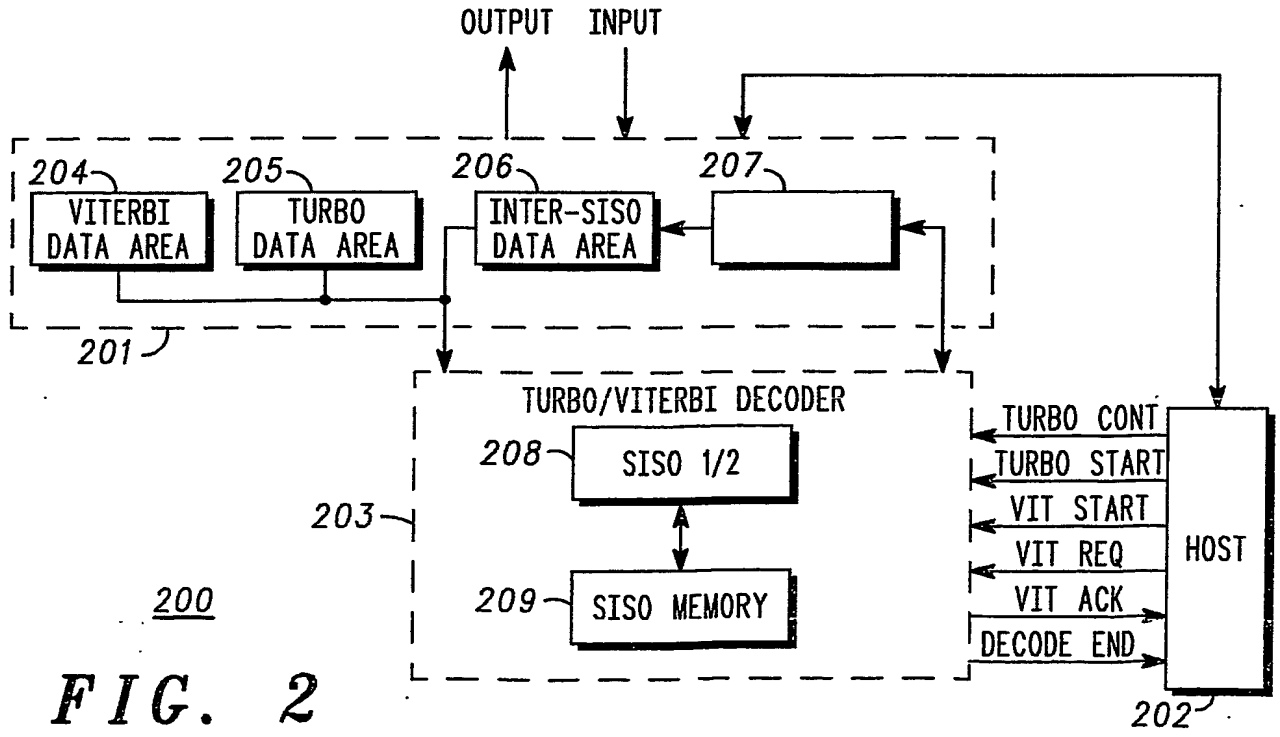

OUTPUT
INPUT
204
VITERBI DATA AREA
205
TURBO DATA AREA
206
INTER-SISO DATA AREA
207
201
TURBO/VITERBI DECODER
208
SISO 1/2
203
209
SISO MEMORY
200
TURBO CONT
TURBO START
VIT START
VIT REQ
VIT ACK
DECODE END
HOST
202

FIG. 2

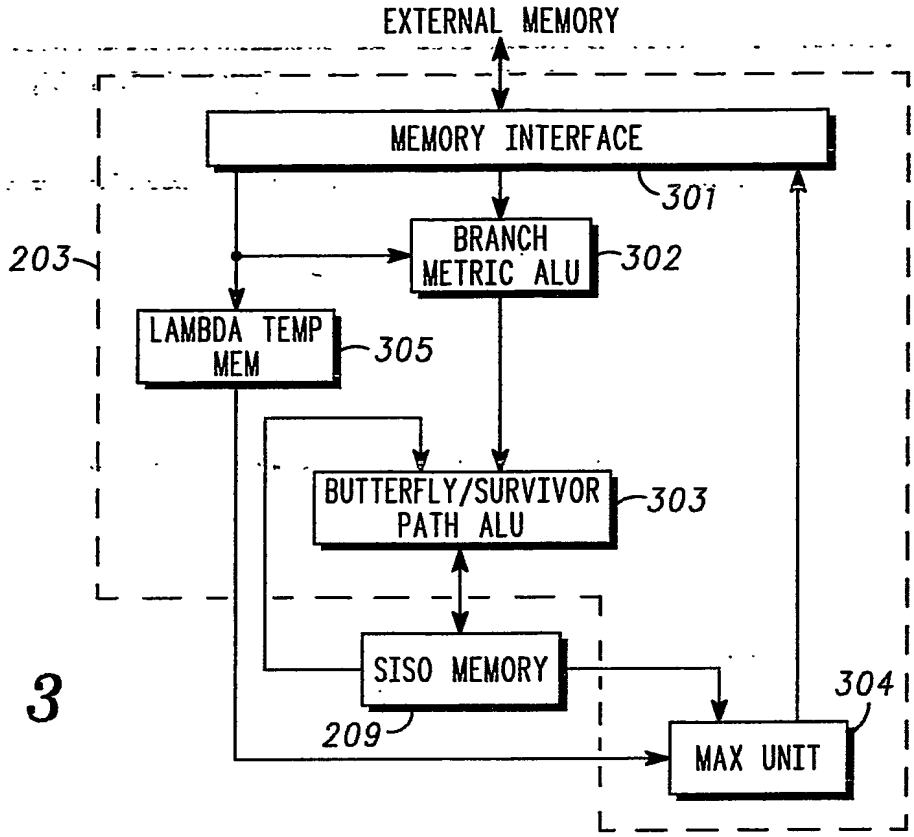


FIG. 3

DATA STORAGE FOR VITERBI DECODING

	16 BYTES	16 BYTES
PART A	OLD P.M. AND S.P. 0-3	OLD P.M. AND S.P. 4-7
	⋮	⋮
	OLD P.M. AND S.P. 120-123	OLD P.M. AND S.P. 124-127
	OLD P.M. AND S.P. 132-135	OLD P.M. AND S.P. 128-131
	⋮	⋮
	OLD P.M. AND S.P. 252-255	OLD P.M. AND S.P. 248-251
PART B	NEW P.M. AND S.P. 0-3	NEW P.M. AND S.P. 4-7
	⋮	⋮
	NEW P.M. AND S.P. 120-123	NEW P.M. AND S.P. 124-127
	NEW P.M. AND S.P. 132-135	NEW P.M. AND S.P. 128-131
	⋮	⋮
	NEW P.M. AND S.P. 252-255	NEW P.M. AND S.P. 248-251
	MEMORY GROUP A	MEMORY GROUP B

FIG. 4

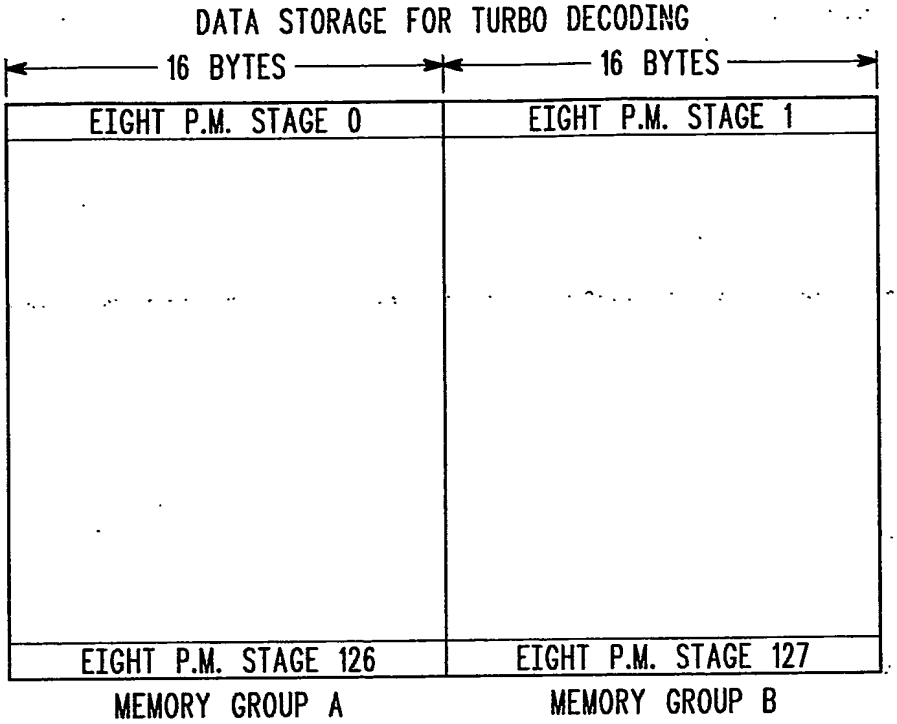

DATA STORAGE FOR TURBO DECODING
16 BYTES
16 BYTES
EIGHT P.M. STAGE 0
EIGHT P.M. STAGE 1
EIGHT P.M. STAGE 126
EIGHT P.M. STAGE 127
MEMORY GROUP A
MEMORY GROUP B

FIG. 5